

C2115

Praktický úvod do superpočítání

IX. lekce

Petr Kulhánek

kulhanek@chemi.muni.cz

Národní centrum pro výzkum biomolekul, Přírodovědecká fakulta
Masarykova univerzita, Kamenice 5, CZ-62500 Brno

Dávkové systémy (začínáme)

Dávkové zpracování

Dávkové zpracování je vykonávání série programů (tzv. dávek) na počítači bez účasti uživatele. Dávky jsou připraveny předem, takže mohou být zpracovány předány bez účasti uživatele. Všechna vstupní data jsou předem připravena v souborech (skriptech) nebo zadána pomocí parametrů na příkazovém řádku. Dávkové zpracování je opakem interaktivního zpracování, kdy uživatel až teprve za běhu programu poskytuje požadované vstupy.

Výhody dávkového zpracování

- sdílení zdrojů počítače mezi mnoha uživateli a programy
- odložení zpracování dávek do doby, kdy je počítač méně vytížen
- odstranění prodlev způsobeným čekáním na vstup od uživatele
- maximalizace využití počítače zlepšuje využití investic (zejména u dražších počítačů)

zdroj: www.wikipedia.cz, upraveno

Nástroje pro dávkové spouštění

➤ **OpenPBS**

<http://www.mcs.anl.gov/research/projects/openpbs/>

➤ **Oracle Grid Engine**

<http://www.oracle.com/us/products/tools/oracle-grid-engine-075549.html>

➤ **Open Grid Scheduler**

<http://gridscheduler.sourceforge.net/>

open source

➤ **Torque**

<http://www.adaptivecomputing.com/products/open-source/torque/>

➤ **PBSPro**

<http://www.pbsworks.com>, <http://pbspro.org/>

je použit jako dávkový systém na našich lokálních klastrech, v MetaCentrum VO, i IT4I

PBSPro

Dokumentace:

<http://www.pbsworks.com/pdfs/PBSProUserGuide13.1.pdf>

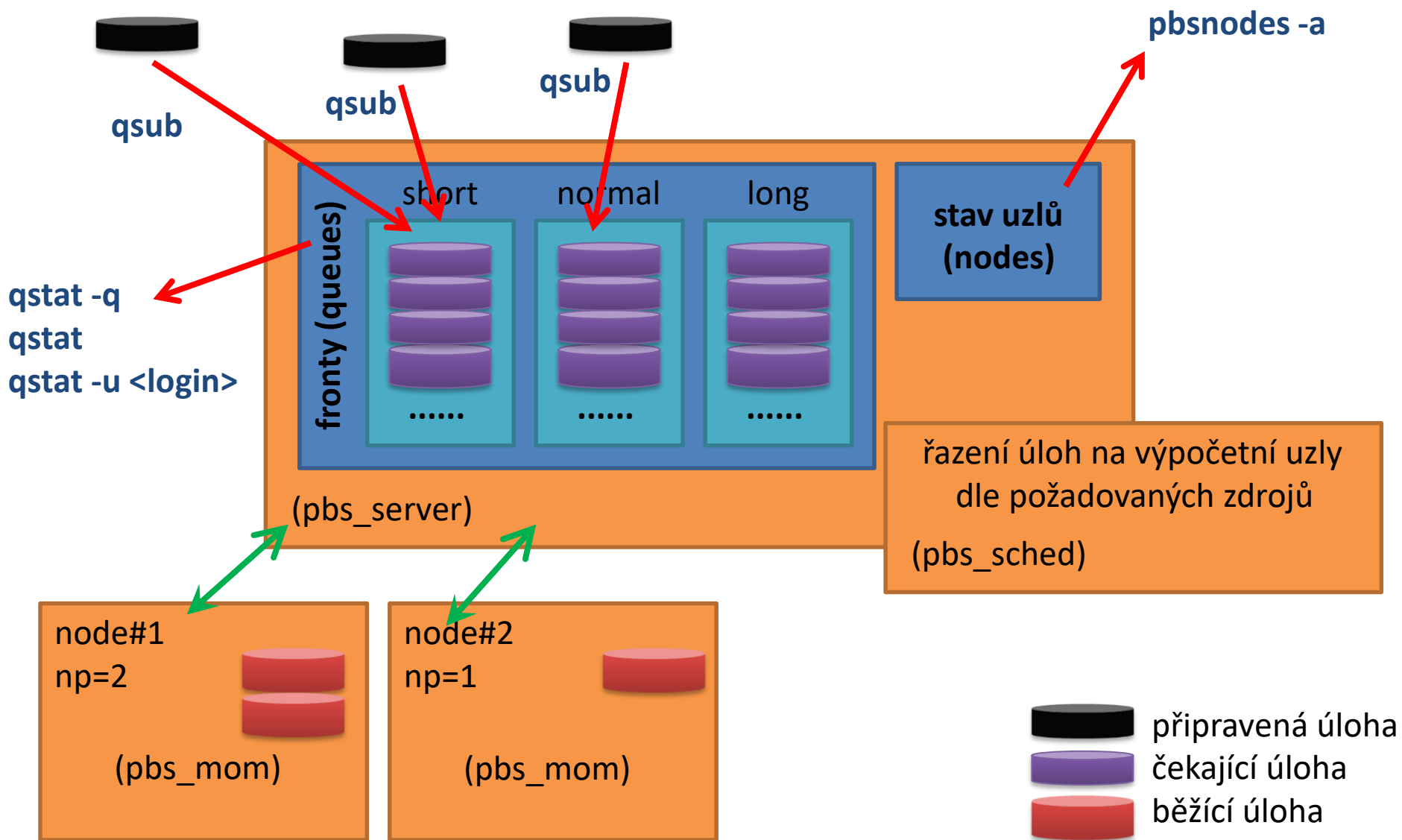
Nezbytná podmínka

Přihlašování bez hesla

Mezi výpočetními uzly a serverem (a naopak) je nutné nastavit přihlašování pomocí ssh bez explicitního zadávání hesla.

- **naše lokální klastry (WOLF, ...), IT4I** – pomocí ssh klíčů
- **MetaCentrum** – v době zasílání úlohy příkazem qsub do dávkového systému musíte mít platný kerberovský lístek

Architektura



Torque – příkazy, stavy úlohy

qsub	zašle úlohu do dávkového systému
qstat	vypíše informace o dávkovém systému (seznam úloh, seznam front)
pbsnodes	vypíše informace o výpočetních uzlech
qrls	uvolní úlohu ze stavu holded (pokud to okolnosti dovolují)

Stavy úlohy:

Q (queued)	čeká ve frontě na spuštění na výpočetním uzlu
R (running)	běží na výpočetním uzly
C (completed)	úloha byla dokončena (informace o dokončených úloh se zobrazují jen omezenou dobu – nejčastěji 24 hodin)
H (holded)	úloha byla pozastavena, úlohu je možné uvolnit příkazem qrls
E (error)	došlo k chybě

Zadááme úlohy

K zadávání úloh do dávkového systému se používá příkaz **qsub**.

```
$ qsub -q default uloha.sh  
1.ubuntu
```

jméno fronty, do které
chceme úlohu zařadit

skript úlohy, např.

```
#!/bin/bash  
echo "Hello world from `hostname`!"
```

příkaz vypíše ID úlohy, pokud
zařazení proběhne v pořádku

```
$ ls  
uloha.sh uloha.sh.o1 uloha.sh.e1
```

soubory jsou
dostupné až po
skončení úlohy

standardní výstup
úlohy

standardní chybový
výstup úlohy

Cvičení 1

1. Jaké fronty dávkového systému jsou dostupné na klastru WOLF. Použijte příkaz `qstat` a volbu dle dokumentace.
2. Jaký je rozdíl mezi volbou `-Q` a `-q` příkazu `qstat`?
3. Jaké úlohy jsou již zařazeny v dávkovém systému klastru WOLF?
4. Skript úlohy z předchozí ukázky vložte do samostatného adresáře a zařadte jej do dávkového systému. Použijte frontu `default`.
5. Na jakém výpočetním uzlu se úloha spustila?
6. Vytvořte nový skript úlohy, který umístíte do jiného adresáře. Skript vypíše jméno výpočetního uzlu a pozastaví svou činnost na 10 minut. Úlohu zařadte do fronty `default`.
7. Monitorujte stav dávkového systému příkazy `qstat` a `pbsnodes`.
8. Na kterém výpočetním uzlu se úloha spustila tentokrát?

Cvičení 2

1. Přihlaste se na čelní uzel MetaCentra `onyx.ncbr.muni.cz`.
2. Jaké fronty dávkového systému jsou dostupné. Použijte příkaz `qstat`.
3. Kolik úloh je aktuálně zařazeno do dávkového systému?
4. Skript úlohy z předchozí ukázky vložte do samostatného adresáře a zařadte jej do dávkového systému. Použijte frontu `default`.
5. Na jakém výpočetním uzlu se úloha spustila? Jak dlouho trvalo, než se úloha spustila?

Cvičení 3

1. Přihlaste se na čelní uzel MetaCentra `zuphux.cerit-sc.cz`.
2. Jaké fronty dávkového systému jsou dostupné. Použijte příkaz `qstat`. Proč se liší od front, které jste viděli na čelním uzlu `onyx.ncbr.muni.cz`?
3. Kolik úloh je aktuálně zařazeno do dávkového systému?
4. Skript úlohy z předchozí ukázky vložte do samostatného adresáře a zařadte jej do dávkového systému. Použijte frontu `default`. V čem se liší identifikátor úlohy od identifikátoru úlohy zařazené na čelním uzlu `onyx.ncbr.muni.cz`?
5. Na jakém výpočetním uzlu se úloha spustila? Jak dlouho trvalo, než se úloha spustila?

Alokace zdrojů

https://wiki.metacentrum.cz/wiki/PBS_Professional

zdroje se zadávají pomocí volby **-l** příkazu `qsub`, lze zadat více specifikací současně např:

```
$ qsub -l select=1:ncpus=1:mem=400mb:scratch_local=10gb \  
skript.sh
```

nebo

```
$ qsub -l select=1:ncpus=1:mem=400mb:scratch_local=10gb \  
-l walltime=10:00 skript.sh
```

Počet a typ výpočetních uzlů a CPU

```
select=[N1] chunk_specification1 [+ [N2] chunk_specification1]
```



počet bloků (chunks)

Příklad:

```
select=1 : ncpus=1 : mem=400mb : scratch_local=10gb
```

specifikace bloku

Slouží pouze k rezervaci výpočetních zdrojů. To však neznamená, že úloha na přidělených výpočetních zdrojích bude automaticky spuštěna. Toto musí zajistit skript úlohy.

Poččet a typ výpočetních uzlů a CPU, II

Seznam alokovaných CPU je dostupný jako seznam výpočetních uzlů uvedených v souboru, jehož název je uveden v systémové proměnné **PBS_NODEFILE**. Tato proměnná je dostupná v běžící úloze:

```
#!/bin/bash
echo $PBS_NODEFILE
cat $PBS_NODEFILE
```

Příklad:

```
$ qsub -l select=1:ncpus=2+1:ncpus=1 skript.sh
```

Výsledek:

```
/var/spool/torque/aux//10312644.arien-pro.ics.muni.cz
zubat2.ncbr.muni.cz
zubat2.ncbr.muni.cz
mandos2.ics.muni.cz
```

Seznam CPU slotů je pak dostupný v úplném popisu úlohy, položka **exec_host**:

```
$ qstat -f <cislo_ulohy>
```

Počet a typ výpočetních uzlů a CPU, III

Vlastnosti:

Výpočetní uzly mohou mít specifikované vlastnosti. Jedná se o krátké řetězce, jejichž význam je závislý na administrátorech systému. Vlastnosti uzlů jsou vypisovány příkazem `pbsnodes` položka `resources_available`.

Uživatel může ve specifikaci výpočetních zdrojů požadovat pouze takové výpočetní uzly, které mají specifikované vlastnosti.

Příklady:

```
select=1:ncpus=1:brno=True
```

```
select=1:ncpus=1:os=debian80
```

```
select=1:ncpus=1:vnode=zubat1
```

```
select=1:ncpus=1:vnode=^zubat1
```


exclusion

Další specifikace zdrojů

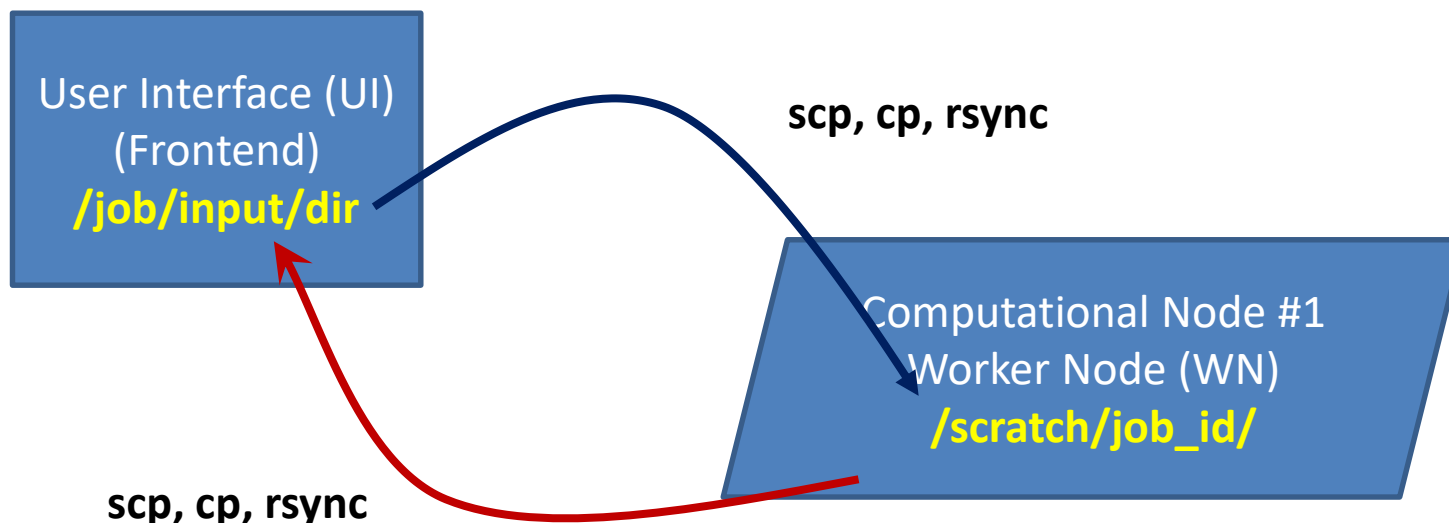
Zdroj	Popis
mem	velikost paměti, jednotky mb, gb
scratch_local	velikost lokálního datového úložiště, jednotky mb, gb
scratch_ssd	velikost lokálního datového úložiště na SSD, jednotky mb, gb
walltime	maximální doba běhu úlohy ve spojení s frontou default

V MetaCentru mohou být úlohy s nedostatečně specifikovanými požadavky na zdroje předčasně ukončeny.

Kopírování souborů

Kopírování souborů

Torque má vnitřní podporu pro kopírování souborů pomocí `stagein` a `stageout` direktiv. Tento způsob je však prakticky nepoužitelný a uživatel by měl veškeré operace související s kopírováním dat na lokální datové úložiště zajistit v rámci úlohy (příkazy `cp`, `scp`, `rsync`). Tento způsob je popsán v dokumentaci MetaCentrum VO.



MetaCentrum

Dávkové systémy

MetaCentrum VO se skládá ze dvou oddělených dávkových systémů:

- **arjen-pro.ics.muni.cz** obsluhuje výpočetní uzly z MetaCentra, výchozí na všech čelních uzlech, kromě **zuphux.cert-sc.cz**
- **wagap-pro.cerit-sc.cz** obsluhuje výpočetní uzly z CERIT-SC, výchozí na čelním uzlu **zuphux.cert-sc.cz**

Oba systémy jsou uživatelsky kompatibilní (stejně volby), rozdíly je možné najít v dokumentaci MetaCentrum VO.

Výchozí Torque server lze změnit nastavením proměnné PBS_SERVER, např.

```
[kulhanek@zuphux ~]$ qstat  
[kulhanek@zuphux ~]$ export PBS_SERVER=arjen-pro.ics.muni.cz  
[kulhanek@zuphux ~]$ qstat
```

← vypíše úlohy z CERIT-SC

← vypíše úlohy z MetaCentra

Spouštění programu gaussian v MetaCentru

Cvičení 4

Cílem cvičení je vytvořit model molekuly C_{60} a vypočítat její molekulární vibrace semiempirickou kvantově-chemickou metodou PM6 v programu gaussian verze 09.

Výsledek následujícího cvičení uvádějte do protokolu souhrnně, uvádějte pouze důležité informace.

1. Do programu Nemesis načtěte strukturu molekuly C_{60} (File → Import structure from → XYZ).
2. Vytvořte vstupní soubor pro program gaussian (File → Export Structure as ... → Gaussian Input). Zvolte metodu PM6 a optimalizaci geometrie. Poté do vstupního souboru dopište klíčové slovo **FREQ** (za klíčové slovo **Opt**) a soubor uložte s příponou **.com**.
3. Vytvořený vstupní soubor přeneste na čelní uzel MetaCentra, připravte skript úlohy a úlohu zařadte do dávkového systému. Postupujte podle dokumentace MetaCentra, **úloha musí na výpočetním uzlu používat lokální datové úložiště.**
4. Výsledek úlohy (soubor se zakončením **.log**) přeneste na vaši pracovní stanici a vypočítané molekulární vibrace zobrazte v programu Nemesis, podle návodu uvedenému dále.

Nemesis

Spuštění programu:

```
$ module add nemesis  
$ nemesis
```

Myš:

Levé tlačítko	selekce
Prostřední tlačítko	rotace
Levé tlačítko	posun
Kolečko	zoom

Modifikátory:

Shift	XZ -> Y pohyby
Ctrl	přepíná mezi sekundárním a primárním manipulátorem

Build Project

The screenshot shows the NEMESIS Molecular Modelling Package interface. The main window is titled "Project 1 : NEMESIS - Molecular Modelling Package". The interface is divided into several panels:

- Structures panel:** Contains a table with columns "Name", "SID", and "Ato". It lists "Structure 1" with SID "1". A blue arrow points from the text "vrstvy" to the "Structure 1" entry.
- Build panel:** Contains chemical symbols for atoms and bonds (C, F, O, N, Cl, Br, S, I) and buttons for "Delete atom", "Make bond", "Break bond", "Delete bond", and "Optimize". A blue arrow points from the text "stavba/editace molekuly" to the "Optimize" button, which is circled in red.
- Geometry panel:** Contains buttons for "Position", "Distance", "Angle", and "Torsion". A blue arrow points from the text "měření geometrie" to this panel.
- Profile objects panel:** Contains a table with columns "Name" and "Ty". It lists "Light 1", "Background 1", "Standard Model 1", and "Freezed Atoms 1". A blue arrow points from the text "grafické modely" to this panel.

Other annotations include "optimalizace geometrie pomocí silového pole" with a blue arrow pointing to the "Optimize" button.

Nastavení silového pole pro optimalizaci: menu Geometry-> Optimizer Setup

Trajectory: Vizualizace vibrací

- 1) Projekt: Trajectory
- 2) File->Import Trajectory as -> Gaussian Vibrations

The screenshot shows the NEMESIS Molecular Modelling Package interface. The main window displays a ball-and-stick model of an ethane molecule. A blue arrow labeled "dvojklik" (double-click) points from the "Trajectory 1 Structure 1" entry in the "Trajectories" panel to the "Trajectory" dialog box. The "Trajectory" dialog has tabs for "Basic", "Play", "Segments", "Filters", and "Referenced by". It contains a table with the following data:

SI	Name	Snapshots	Type
1	ethan_freq	180	Gaussian Vibratic

Another blue arrow labeled "dvojklik" points from the "ethan_freq" entry to the "Gaussian Vibrations" dialog box. This dialog has tabs for "Basic", "Vibrations", and "Setup". The "Vibrations" tab contains a table with the following data:

ID	Frequency	IR Intensity	Scale
1	224.6	0.0	
2	878.2	0.0	
3	878.2	0.0	
4	1120.0	0.0	
5	1120.0	0.0	
6	1137.8	0.0	
7	1359.3	0.0	
8	1408.2	0.0	
9	1408.2	0.0	
10	1443.6	0.0	

Below the table are fields for "Number of vibrations:" (set to 24) and "Active vibrations:" (set to 0), along with "Activate imaginary" and "Deactivate all" buttons. A blue arrow labeled "zvolíme vibraci" (we will choose vibrations) points to the "ID" column of the table. Another blue arrow labeled "spustíme animaci" (we will start animation) points to the play button in the bottom control bar. The control bar also shows a progress indicator from 1 to 180 snapshots.